

Sym

PAS

PPPPPPPPPPPPPP
PPPPPPPPPPPPPP
PPPPPPPPPPPPPP
PPP PPP AAA AAA SSS RRR RRR TTT LLL
PPP PPP AAA AAA SSS RRR RRR TTT LLL
PPP PPP AAA AAA SSS RRR RRR TTT LLL
PPP PPP AAA AAA SSS RRR RRR TTT LLL
PPP PPP AAA AAA SSS RRR RRR TTT LLL
PPP PPP AAA AAA SSS RRR RRR TTT LLL
PPP PPP AAA AAA SSS RRR RRR TTT LLL
PPPPPPPPPPPPPP
PPPPPPPPPPPPPP
PPPPPPPPPPPPPP
AAA AAA SSSSSSSSS RRRRRRRRRRRR TTT LLL
AAA AAA SSSSSSSSS RRRRRRRRRRRR TTT LLL
AAA AAA SSSSSSSSS RRRRRRRRRRRR TTT LLL
PPP AAA AAA SSS RRR RRR TTT LLL
PPP AAA AAA SSSSSSSSSSS RRR RRR TTT LLL
PPP AAA AAA SSSSSSSSSSS RRR RRR TTT LLL
PPP AAA AAA SSSSSSSSSSS RRR RRR TTT LLL

FILE ID**PASUNLOCK

07

PAS
1-0

```

PPPPPPPP    AAAAAAA SSSSSSSS UU   UU NN   NN LL   000000   CCCCCCCC KK   KK
PPPPPPPP    AAAAAAA SSSSSSSS UU   UU NN   NN LL   000000   CCCCCCCC KK   KK
PP          PP AA   AA SS   UU   UU NN   NN LL   00   00 CC   KK   KK
PP          PP AA   AA SS   UU   UU NNNN  NN LL   00   00 CC   KK   KK
PP          PP AA   AA SS   UU   UU NNNN  NN LL   00   00 CC   KK   KK
PP          PP AA   AA SS   UU   UU NNNN  NN LL   00   00 CC   KK   KK
PPPPPPPP    AA   AA SSSSSS UU   UU NN   NN LL   00   00 CC   KKKKKK
PPPPPPPP    AA   AA SSSSSS UU   UU NN   NN LL   00   00 CC   KKKKKK
PP          AAAAAAAA SS   UU   UU NN   NNNN LL   00   00 CC   KK   KK
PP          AAAAAAAA SS   UU   UU NN   NNNN LL   00   00 CC   KK   KK
PP          AA   AA SS   UU   UU NN   NN LL   00   00 CC   KK   KK
PP          AA   AA SS   UU   UU NN   NN LL   00   00 CC   KK   KK
PP          AA   AA SSSSSSSS UUUUUUUUUU NN   NN LLLLLLLL 000000   CCCCCCCC KK   KK
PP          AA   AA SSSSSSSS UUUUUUUUUU NN   NN LLLLLLLL 000000   CCCCCCCC KK   KK

```

The diagram illustrates the assembly of a 3D structure using two types of blocks: L-shaped blocks (represented by 'L' characters) and S-shaped blocks (represented by 'S' characters). The structure is built in layers, starting with a single L-shaped block at the bottom left. Subsequent layers are added to the right and top of the previous layer, following a specific pattern. By the final stage, a large vertical column of L-shaped blocks is visible on the left, while the right side features a series of horizontal layers composed of S-shaped blocks.

```
: 1 0001 0 MODULE PAS$UNLOCK ( %TITLE 'UNLOCK procedure'  
: 2 0 IDENT = '1-001'  
: 3 0 ) =  
: 4 1 BEGIN  
: 5 1  
: 6 1 *****  
: 7 1 *  
: 8 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
: 9 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
: 10 1 * ALL RIGHTS RESERVED.  
: 11 1 *  
: 12 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
: 13 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
: 14 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
: 15 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
: 16 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
: 17 1 * TRANSFERRED.  
: 18 1 *  
: 19 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
: 20 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
: 21 1 * CORPORATION.  
: 22 1 *  
: 23 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
: 24 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
: 25 1 *  
: 26 1 *  
: 27 1 *****  
: 28 1 :  
: 29 1 :  
: 30 1 ++  
: 31 1 : FACILITY: Pascal Language Support  
: 32 1 :  
: 33 1 : ABSTRACT:  
: 34 1 :  
: 35 1 : This module contains PAS$UNLOCK, which implements the  
: 36 1 : VAX-11 Pascal UNLOCK procedure.  
: 37 1 :  
: 38 1 : ENVIRONMENT: User mode - AST reentrant  
: 39 1 :  
: 40 1 : AUTHOR: Steven B. Lionel, CREATION DATE: 29-January-1982  
: 41 1 :  
: 42 1 : MODIFIED BY:  
: 43 1 :  
: 44 1 : 1-001 - Original. SBL 29-January-1982  
: 45 1 :--  
: 46 1 :
```

```
: 48      0047 1 %SBTTL 'Declarations'  
49      0048 1  
50      0049 1 : PROLOGUE DEFINITIONS:  
51      0050 1  
52      0051 1  
53      0052 1 REQUIRE 'RTLIN:PASPROLOG';           ! Externals, linkages, PSECTs, structures  
54      0116 1  
55      0117 1  
56      0118 1 : TABLE OF CONTENTS:  
57      0119 1  
58      0120 1  
59      0121 1 FORWARD ROUTINE  
60      0122 1     PASSUNLOCK: NOVALUE;          ! Do a UNLOCK  
61      0123 1  
62      0124 1  
63      0125 1 : MACROS:  
64      0126 1  
65      0127 1     NONE  
66      0128 1  
67      0129 1 : EQUATED SYMBOLS:  
68      0130 1  
69      0131 1     NONE  
70      0132 1  
71      0133 1 : FIELDS:  
72      0134 1  
73      0135 1     NONE  
74      0136 1  
75      0137 1 : OWN STORAGE:  
76      0138 1  
77      0139 1     NONE
```

```
79      0140 1 ZSBTTL 'PASSUNLOCK - UNLOCK procedure'
80      0141 1 GLOBAL ROUTINE PASSUNLOCK (
81          0142 1     PFV: REF $PAS$PFV_FILE_VARIABLE,
82          0143 1     ERROR
83          0144 1     ): NOVALUE =
84
85          0146 1     ++
86          0147 1     FUNCTIONAL DESCRIPTION:
87
88          0149 1     PASSUNLOCK implements the VAX-11 Pascal UNLOCK procedure. It
89          0150 1     unlocks the current file component.
90
91          0152 1     CALLING SEQUENCE:
92
93          0154 1     CALL PASSUNLOCK (PFV.mr.r [, ERROR.j.r])
94
95          0156 1     FORMAL PARAMETERS:
96
97          0158 1     PFV           - The Pascal File Variable (PFV) passed by reference.
98          0159 1           The structure of the PFV is defined in PASPFV.REQ.
99
100         0160 1     ERROR          - Optional. If specified, the address to unwind to
101         0162 1           if an error occurs.
102
103         0164 1     IMPLICIT INPUTS:
104
105         0166 1     NONE
106
107         0168 1     IMPLICIT OUTPUTS:
108
109         0170 1     NONE
110
111         0172 1     ROUTINE VALUE:
112
113         0174 1     NONE
114
115         0176 1     SIDE EFFECTS:
116
117         0178 1     NONE
118
119         0180 1     SIGNALLED ERRORS:
120
121         0182 1     FILNOTOPE - File not open
122         0183 1     FILNOTINS - File is not in Inspection mode
123         0184 1     ERRDURTRU - Error during UNLOCK
124
125         0186 1
126         0187 1     --
127
128         0188 1
129         0189 2     BEGIN
130
131         0190 2
132         0191 2     LOCAL
133             0192 2     FCB: REF $PAS$FCB_CONTROL_BLOCK,
134             0193 2     PFV_ADDR: VOLATILE,
135             0194 2     UNWIND_ACT: VOLATILE,
136             0195 2     ERROR_ADDR: VOLATILE;
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
```

```
136 0197 2      BIND
137 C198 2          RAB = FCB: REF BLOCK [, BYTE];           ! RMS RAB
138 C199 2
139 200 2      BUILTIN
140 J201 2          ACTUALCOUNT;
141 202 2
142 203 2      ENABLE
143 204 2          PASS$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR);
144 205 2
145 206 2      IF ACTUALCOUNT () GEQU 2
146 THEN
147 207 2          ERROR_ADDR = .ERROR;                      ! Set unwind address
148 208 2
149 209 2
150 210 2
151 211 2      |+
152 212 2          | Set PFV address enable argument.
153 213 2          |-
154 214 2
155 215 2
156 216 2      |+
157 217 2          | Validate and lock PFV
158 218 2          |-
159 219 2
160 220 2      PASS$VALIDATE_PFV (PFV [PFV$R_PFV]; FCB);
161 221 2
162 222 2      |+
163 223 2          | Set unwind action to unlock file.
164 224 2          |-
165 225 2
166 226 2      UNWIND_ACT = PASS$K_UNWIND_UNLOCK;
167 227 2
168 228 2      |+
169 229 2          | Open file if it should be implicitly opened.
170 230 2          |-
171 231 2
172 232 2      IF NOT .PFV [PFV$V_VALID]
173 THEN
174 233 2          PASS$OPEN_IMPLICT (PFV [PFV$R_PFV], FCB [FCB$R_FCB]; FCB);
175 234 2
176 235 2      |+
177 236 2          | Verify that the file is open.
178 237 2          |-
179 238 2
180 239 2
181 240 2      IF NOT .PFV [PFV$V_OPEN]
182 THEN
183 241 2          SPASSIO_ERROR (PASS$_FILNOTOPE,0);
184 242 2
185 243 2      |+
186 244 2          | Verify that the file is in Inspection mode.
187 245 2          |-
188 246 2
189 247 2
190 248 2      IF NOT .FCB [FCB$V_INSPECTION]
191 THEN
192 249 2          SPASSIO_ERROR (PASS$_FILNOTINS,0);
193 250 2
194 251 2
195 252 2
196 253 2      |+
197 254 2          | Do the $RELEASE to unlock the current component.
```

```

193      0254 2      !-
194      0255 2
195      0256 3      IF NOT SPASSRMS_OP ($RELEASE (RAB=.RAB))
196      0257 3      THEN
197      0258 3      BEGIN
198      0259 3      !+
199      0260 3      $RELEASE failed. Determine if the error was "record not locked".
200      0261 3      If so, ignore the error.
201      0262 3      !-
202      0263 3
203      0264 3      IF .RAB [RAB$L_STS] NEQU RMSS_RNL
204      0265 3      THEN
205      0266 3      SPASSIO_ERROR (PASS_ERRDURUNL);    ! Error during UNLOCK
206      0267 2      END;
207      0268 2
208      0269 2      !+
209      0270 2      Cancel previous LOCATE
210      0271 2      Indicate successful operation
211      0272 2      Unlock the file
212      0273 2      !-
213      0274 2
214      0275 2      FCB [FCB$L_STATUS] = 0;
215      0276 2      FCB [FCB$V_LOCATE] = 0;
216      0277 2      PFV [PFV$V_LOCK] = 0;
217      0278 2
218      0279 2      RETURN;
219      0280 2
220      0281 1      END:                                ! End of routine PASSUNLOCK

```

.TITLE PASSUNLOCK UNLOCK procedure
.IDENT \1-001\

.EXTRN PASSUNLOCK, PASS\$IO_HANDLER
.EXTRN PASS\$VALIDATE_PFV
.EXTRN PASS\$OPEN_IMPPLICIT
.EXTRN PASS\$SIGNAL, PASSK_FILENOPE
.EXTRN PASSK_FILENOINS
.EXTRN SYSSRELEASE, PASSK_ERRDURUNL
.PSECT _PASSCODE,NOWRT, SHR, PIC,2

<pre> 53 0000000G 00 00CC 00000 5E 00 9E 00002 08 C2 00009 7E D4 0000C 6D 04 AE 7C 0000E 007D CF DE 00011 02 6C 91 00016 04 1F 00019 6E 08 AC D0 0001B 04 AE D0 0001F 1\$: 08 56 56 D0 00023 04 AE 0000000G 00 16 00027 01 D0 0002D 06 A6 E8 00031 0000000G 00 16 00035 </pre>	<pre> .ENTRY PASSUNLOCK, Save R2,R3,R6,R7 MOVAB PASS\$SIGNAL, R3 SUBL2 #8, SP CLRL ERROR_ADDR CLRQ UNWIND_ACT MOVAL 8\$, (FP) CMPB (AP), #2 BLSSU 1\$: MOVL ERROR, ERROR_ADDR MOVL PFV, R6 MOVL R6, PFV_ADDR JSB PASS\$VALIDATE_PFV MOVL #1, UNWIND_ACT BLBS 6(R6), 2\$- JSB PASS\$OPEN_IMPLICIT </pre>	0141 0189 0206 0208 0214 0220 0226 0232 0234
---	---	--

PASSUNLOCK
1-001UNLOCK procedure
PASSUNLOCK - UNLOCK procedureJ 7
16-Sep-1984 02:11:26
14-Sep-1984 12:51:58
VAX-11 Bliss-32 V4.0-742
[PASRTL.SRC]PASUNLOCK.B32;1Page 6
(3)

PA

08	07	A6		05	E0 0003B 2\$:	BBS	#5, 7(R6), 3\$		0240
		7E	00G	7E	D4 00040	CLRL	-(SP)		0242
				8F	9A 00042	MOVZBL	#PASSK_FILNOTOPE, -(SP)		
0A	FD	A7		OB	11 00046	BRB	4\$		0248
				03	E0 00048 3\$:	BBS	#3, -3(FCB), 5\$		0250
		7E	00G	7E	D4 0004D	CLRL	-(SP)		
			63	8F	9A 0004F	MOVZBL	#PASSK_FILNOTINS, -(SP)		
				02	FB 00053 4\$:	CALLS	#2, PASS\$SIGNAL		
				04	00056	RET			
00000000G	00			57	DD 00057 5\$:	PUSHL	FCB		0256
	22			01	FB 00059	CALLS	#1, SYSSRELEASE		
0001825A	8F			50	E8 00060	BLBS	SSSTATUS, 7\$		
				50	D1 00063	CMPL	SSSTATUS, #98906		
		E7		04	12 0006A	BNEQ	6\$		
000181A0	12		FF	A7	E8 0006C	BLBS	-1(FCB), 5\$		
	8F			50	E8 00070 6\$:	BLBS	SSSTATUS, 7\$		0264
		08		A7	D1 00073	CMPL	8(FCB), #98720		
				08	13 0007B	BEQL	7\$		
		7E	00G	8F	9A 0007D	MOVZBL	#PASSK_ERRDURUNL, -(SP)		0266
			63	01	FB 00081	CALLS	#1, PASS\$SIGNAL		
				04	00084	RET			
		FE	A7	04	A7 D4 00085 7\$:	CLRL	-44(FCB)		0275
	07	A6		01	8A 00088	BICB2	#1, -2(FCB)		0276
			80	8F	8A 0008C	BICB2	#128, 7(R6)		0277
				04	00091	RET			0281
				0000	00092 8\$:	.WORD	Save nothing		0189
			50	08	AC D0 00094	MOVL	8(AP), R0		
			50	04	A0 D0 00098	MOVL	4(R0), R0		
				F4	A0 9F 0009C	PUSHAB	ERRQR_ADDR		
				F8	A0 9F 0009F	PUSHAB	UNWIND ACT		
				FC	A0 9F 000A2	PUSHAB	PFV_ADDR		
				03	DD 000A5	PUSHL	#3		
				SE	DD 000A7	PUSHL	SP		
00000000G	7E	00	04	AC	7D 000A9	MOVQ	4(AP), -(SP)		
			03	FB C00AD	CALLS	#3, PASS\$IO_HANDLER			
			04	000B4	RET				

: Routine Size: 181 bytes, Routine Base: _PASS\$CODE + 0000

: 221

0282 1

: 222

0283 1 !<BLF/PAGE>

PASSUNLOCK
1-001 UNLOCK procedure
 PASSUNLOCK - UNLOCK procedure
.: 224 0284 1 END
.: 225 0285 1
.: 226 0286 0 ELUDOM

K 7
16-Sep-1984 02:11:26 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:51:58 [PASRTL.SRC]PASUNLOCK.B32;1

Page 7
(4)

PA
1-

: End of module PASSUNLOCK

PSECT SUMMARY

Name	Bytes	Attributes
_PASSCODE	181	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Symbols -----	Total	Loaded	Percent	Pages Mapped	Processing Time
\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	427	7	0	581	00:01.0
\$255\$DUA28:[PASRTL.OBJ]PASLIB.L32;1			93	21	33	00:00.4

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS\$:\$PASUNLOCK/OBJ=OBJ\$:\$PASUNLOCK MSRC\$:\$PASUNLOCK/UPDATE=(ENH\$:\$PASUNLOCK)

Size: 181 code + 0 data bytes
Run Time: 00:05.8
Elapsed Time: 00:22.4
Lines/CPU Min: 2953
Lexemes/CPU-Min: 13838
Memory Used: 87 pages
Compilation Complete

0297 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

